UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/955,961	09/20/2001	Gregory S. Andre	017750-416	1901

21839 7590 04/26/2006

BUCHANAN INGERSOLL PC
(INCLUDING BURNS, DOANE, SWECKER & MATHIS)
POST OFFICE BOX 1404
ALEXANDRIA, VA 22313-1404

EXAMINER	
LEE, CHRISTOPHER E	
ART UNIT	**PAPER NUMBER**
2112	

DATE MAILED: 04/26/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/955,961	ANDRE, GREGORY S.	
	Examiner	Art Unit	
	Christopher E. Lee	2112	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *15 March 2006*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-31* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-31* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)

2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date *3/24/06*.

4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .

5) ☐ Notice of Informal Patent Application (PTO-152)

6) ☐ Other: _____.

DETAILED ACTION

Receipt Acknowledgement

1. Receipt is acknowledged of the Amendment filed on 15th of March 2006. Claims 1, and 20-24 have been amended; no claim has been canceled; and no claim has been newly added since the RCE Non-Final Office Action was mailed on 15th of December 2005. Currently, claims 1-31 are pending in this Application.

Claim Rejections - 35 USC § 103

2. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

3. Claims 1-7 and 14-18 are rejected under 35 U.S.C. 103(a) as being unpatentable over Kimura [JP 409022380 A] in view of Dupont [US 6,842,800 B2], Yamagami et al. [JP 408272756 A; hereinafter Yamagami] and Lane et al. [US 5,502,718 A; hereinafter Lane].

Referring to claim 1, Kimura discloses an apparatus (i.e., multilevel bus connection type multiprocessor system) for managing flow of information among plural processors of a processing array (See Abstract), comprising:

- a plurality of processors (i.e., processors 3 in a plurality of processor modules 1 in Fig. 1), each processor being in communication with a respective local processor bus (i.e., each processor 3 being in communication with a respective bus in module 5 in Fig. 1);
- a system bus (i.e., system bus 2 of Fig. 1) for interconnecting at least two processors for providing a path (i.e., an interconnecting path for processor modules 1, ultimately interconnecting processors 3 of said processor modules in Fig. 1).

Kimura does not teach said path being for packets of data and control information, control information packets being separately buffered from other packets in a buffer separately connected between the respective local processor bus and the system bus.

Dupont discloses a system for managing configurable buffer size (See Abstract), wherein

- a path for packets (i.e., Internet connection) of data and control information (i.e., TCP/IP data and control packets; See col. 4, lines 24-29), control information packets (i.e., TCP/IP control packets) being separately buffered from other packets in a buffer (i.e., smaller buffer unit size buffer subsection 90 of Fig. 2; See col. 4, lines 29-38) separately connected between a respective local processor bus and a system bus (See col. 4, lines 16-23; actually, two separate buffer storage sections 90, 100 in Fig. 2 being connected between input queue 20 and output queue 30 in Fig. 1, wherein said input queue being coupled to Internet network and said output queue being coupled to Internet Access local bus, as an example).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to have included said configurable buffers, as disclosed by Dupont, in said apparatus for managing flow of information, as disclosed by Kimura, so as not to store said control information packet (i.e., small control packet) in large buffer units, and also as not to segment said data packet (i.e., large data packet) into a plurality of smaller segments for storage in smaller buffer units, for the advantage of providing an improved buffering by way of using multiple buffer unit sizes for storing packets of said data and control information (i.e., data or traffic; See Dupont, col. 1, lines 52-53 and col. 4, lines 38-42).

Kimura, as modified by Dupont, does not teach means for arbitrating access to at least a first portion of said system bus among said at least two processors to transfer said packets of data and control information over said first portion, said packets being transferred using a

protocol by which said system bus performs control actions for system bus access independently of said at least two processors.

Yamagami discloses a multiprocessor system (Fig. 1), wherein

- means for arbitrating (i.e., a system bus arbitrating mechanism 3 in Fig. 1; See Abstract on Brief Summary) access to at least a first portion (i.e., a portion of system bus between boot ROM 2 and selected processors 4, 5 and 6 in Fig. 1) of a system bus (i.e., system bus 1 of Fig. 1) among at least two processors (i.e., processors 4, 5 and 6 in Fig. 1) to transfer packets of data and control (i.e., codes in said boot ROM 2 in Fig. 1) over said first portion (See para. [0027]), said packets being transferred using a protocol (i.e., a procedure used to control the orderly exchange of codes between said boot ROM and said selected processor on said system bus) by which said system bus performs control actions for system bus access independently of said at least two processors (See Fig. 2 and paras. [0021]-[0028]).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to have included said means for arbitrating (i.e., system bus mediation device), as disclosed by Yamagami, in said apparatus for managing flow of information, as disclosed by Kimura, as modified by Dupont, for the advantage of providing a method of said apparatus (i.e., multiprocessor system) which can raise the reliability in the case of the boot process of said apparatus (i.e., multiprocessor system; See Yamagami, para. [0008]).

Kimura, as modified by Dupont and Yamagami, does not teach said means for arbitrating establishing a clear path to a destination device by checking device busy signals to ensure that said destination device is not busy.

Lane discloses a device for switching high speed protocol units (See Abstract and Fig. 2), wherein

- means for arbitrating (i.e., arbitration and command module 23 of Fig. 2; See col. 6, lines 37-40) establishing a clear path to a destination device (e.g., matrix interface card 21_N and processing N 25_N in Fig. 2) by checking device busy signals (i.e., checking bus 213_i signal in Fig. 2) to ensure that said destination device is not busy (i.e., ready to receive data; See col. 7, lines 23-37).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to have included said means for arbitrating (i.e., arbitration and command module), as disclosed by Lane, in said apparatus for managing flow of information, as disclosed by Kimura, as modified by Dupont and Yamagami, for the advantage of providing said apparatus, in which the risks of blocking are reduced and in which the duration of these blockages is limited in relation to already known system (See Lane, col. 2, lines 39-42).

Referring to claim 2, Kimura, as modified by Dupont, Yamagami and Lane, teaches

- at least one module (i.e., processor modules 1 in Fig. 1; Kimura) connected by said system bus (i.e., system bus 2 of Fig. 1; Kimura) to said means for arbitrating (i.e., system bus mediation device 3 of Fig. 1; Yamagami).

Referring to claim 3, Kimura, as modified by Dupont, Yamagami and Lane, teaches said at least one module (i.e., processor modules 1 in Fig. 1; Kimura) comprises

- a gateway device (i.e., a first part of system interface controller 6, which is performing interface control between said intramodule bus and said system bus in Fig. 1; Kimura) for communicating via said system bus (i.e., system bus 2 of Fig. 1; See Kimura, para. [0006] step 4) to said means for arbitration (i.e., system bus mediation device 3 of Fig. 1; Yamagami).

Referring to claim 4, Kimura teaches said at least one module (i.e., processor modules 1 in Fig. 1) comprises

- a module bus (i.e., bus in module 5 of Fig. 1) for administering to at least one module node (i.e., a plural pairs of respective processor 3 and proper cache 4 in Fig. 1) within said at least one module (i.e., processor modules).

Referring to claim 5, Kimura teaches said at least one module node (i.e., a plural pairs of respective processor 3 and proper cache 4 in Fig. 1) comprises

- a processing device (i.e., processor 3 in Fig. 1).

Referring to claim 6, Kimura teaches said at least one module node (i.e., a plural pairs of respective processor 3 and proper cache 4 in Fig. 1) comprises

- a bus interface device (i.e., a second part of system interface controller 6, which is performing mediation control of a request on said intramodule bus in Fig. 1) for achieving data communication between said processing device and said module bus (See para. [0006]).

Referring to claim 7, Kimura teaches said at least one module (i.e., processor modules 1 in Fig. 1) comprises

- a local processor bus (i.e., a bus between processor 3 and proper cache 4 in Fig. 1) for communicating data (i.e., cache data) between said processing device (i.e., processor 3 of Fig. 1) and said bus interface device (i.e., said second part of system interface

controller 6 in Fig. 1; in fact, said local processor bus is for communicating cache data between said processor and said second part of system interface controller).

Referring to claim 14, Yamagami teaches

- a system controller (i.e., system bus mediation device 3 of Fig. 1) for controlling access to said system bus (See Fig. 2 and paras. [0021]-[0028]).

Referring to claim 15, Yamagami teaches said system controller (i.e., system bus mediation device 3 of Fig. 1) comprises

- a system bus arbitration unit (i.e., a system bus arbitrating mechanism, disclosed in Abstract on Brief Summary) for controlling access to said system bus (See Fig. 2 and paras. [0021]-[0028]).

Referring to claim 16, Yamagami teaches said system controller (i.e., system bus mediation device 3 of Fig. 1) comprises

- a processor (i.e., processor optional feature 14 of Fig. 1) connected to a bus interface device (i.e., processor holding register 13 of Fig. 1), which is connected to said system bus (i.e., system bus 1 of Fig. 1).

Referring to claim 17, Kimura discloses a method for managing a flow of information among plural processors of a processing array (See Abstract), comprising the steps of

- providing a local connection by a local processor bus (i.e., bus in respective module 5 in Fig. 1) for each of a plurality of processors (i.e., each processor 3 in a plurality of

processor modules 1 being in communication with a respective bus in module 5 in Fig. 1);

- interconnecting at least two processors for providing a path (i.e., an interconnecting path for processor modules 1, ultimately interconnecting processors 3 of said processor modules in Fig. 1) by a system bus (i.e., system bus 2 of Fig. 1).

Kimura does not teach said path being for packets of data and control information, control information packets being separately buffered from other packets in a buffer separately connected between the respective local processor bus and the system bus.

Dupont discloses a method for managing configurable buffer size (See Abstract), wherein

- a path for packets (i.e., Internet connection) of data and control information (i.e., TCP/IP data and control packets; See col. 4, lines 24-29), control information packets (i.e., TCP/IP control packets) being separately buffered from other packets in a buffer (i.e., smaller buffer unit size buffer subsection 90 of Fig. 2; See col. 4, lines 29-38) separately connected between a respective local processor bus and a system bus (See col. 4, lines 16-23; actually, two separate buffer storage sections 90, 100 in Fig. 2 being connected between input queue 20 and output queue 30 in Fig. 1, wherein said input queue being coupled to Internet network and said output queue being coupled to Internet Access local bus, as an example).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to have included said method for managing configurable buffer, as disclosed by Dupont, in said method for managing flow of information, as disclosed by Kimura, so as not to store said control information packet (i.e., small control packet) in large buffer units, and also as not to segment said data packet (i.e., large data packet) into a plurality of smaller segments for storage in smaller buffer units, for the advantage of providing an improved buffering by way of

using multiple buffer unit sizes for storing packets of said data and control information (i.e., data or traffic; See Dupont, col. 1, lines 52-53 and col. 4, lines 38-42).

Kimura, as modified by Dupont, does not teach the step of arbitrating access to at least a first portion of a system bus among said at least two processors to transfer said packets of data and control information over said first portion, said packets being transferred using a protocol by which a system bus performs control actions for system bus access independently of said at least two processors.

Yamagami discloses a multiprocessor system (Fig. 1), wherein

- a step of arbitrating (i.e., a system bus arbitrating mechanism 3 in Fig. 1; See Abstract on Brief Summary) access to at least a first portion (i.e., a portion of system bus between boot ROM 2 and selected processors 4, 5 and 6 in Fig. 1) of a system bus (i.e., system bus 1 of Fig. 1) among at least two processors (i.e., processors 4, 5 and 6 in Fig. 1) to transfer packets of data and control (i.e., codes in said boot ROM 2 in Fig. 1) over said first portion (See para. [0027]), said packets being transferred using a protocol (i.e., a procedure used to control the orderly exchange of codes between said boot ROM and said selected processor on said system bus) by which a system bus performs control actions for system bus access independently of said at least two processors (See Fig. 2 and paras. [0021]-[0028]).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to have included said step of arbitrating (i.e., system bus mediation device), as disclosed by Yamagami, in said method for managing a flow of information, as disclosed by Kimura, as modified by Dupont, for the advantage of providing a method, which can raise the reliability in the case of the boot process of said plural processors of a processing array (i.e., multiprocessor system; See Yamagami, para. [0008]).

Kimura, as modified by Dupont and Yamagami, does not teach that arbitrating access comprises establishing a clear path to a destination device by checking device busy signals to ensure that said destination device is not busy.

Lane discloses a device for switching high speed protocol units (See Abstract and Fig. 2), wherein arbitrating access (See col. 6, lines 37-40) comprises

- establishing a clear path to a destination device (e.g., matrix interface card 21_N and processing N 25_N in Fig. 2) by checking device busy signals (i.e., checking bus 213_i signal in Fig. 2) to ensure that said destination device is not busy (i.e., ready to receive data; See col. 7, lines 23-37).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to have included said arbitrating (i.e., arbitration and command module), as disclosed by Lane, in said method for managing a flow of information, as disclosed by Kimura, as modified by Dupont and Yamagami, for the advantage of providing said apparatus, in which the risks of blocking are reduced and in which the duration of these blockages is limited in relation to already known system (See Lane, col. 2, lines 39-42).

Referring to claim 18, Kimura teaches the step of:

- interconnecting at least one module (i.e., processor modules 1 in Fig. 1) with said system bus (i.e., system bus 2 of Fig. 1) by way of a bus gateway device (i.e., a first part of system interface controller 6, which is performing interface control between said intramodule bus and said system bus in Fig. 1; See para. [0006]), said at least one module (i.e., processor modules) comprising
    - said bus gateway device (i.e., a first part of system interface controller),
    - a module bus (i.e., intramodule bus 5 of Fig. 1),

- at least one processor (i.e., processors 3 in a processor module 1 in Fig. 1), and
- at least one bus interface device (i.e., a second part of system interface controller 6, which is performing mediation control of a request on said intramodule bus in Fig. 1) for connecting said at least one processor to said module bus (See para. [0006] step 4).

4. Claims 8-13 are rejected under 35 U.S.C. 103(a) as being unpatentable over Kimura [JP 409022380 A] in view of Dupont [US 6,842,800 B2], Yamagami [JP 408272756 A] and Lane [US 5,502,718 A] as applied to claims 1-7 and 14-18 above, and further in view of Sand et al. [US 5,990,939 A; hereinafter Sand].

Referring to claims 8, 12, and 13, Kimura, as modified by Dupont, Yamagami and Lane, discloses all the limitations of the claims 8, 12 and 13, respectively, except that does not teach a sensor interface connected to said system bus.

Sand discloses a video demultiplexing interface for a missile tracking system 10 in Fig. 1, wherein

- a sensor interface, which is a forward looking infrared (FLIR) video sensor interface (i.e., Video Thermal Tracker interface 70 in Fig. 2) connected to a system bus (i.e., channel 1, 2, ... N in Fig. 2).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to have combined said sensor interface for said missile tracking system, as disclosed by Sand, with one of said at least two processors (i.e., processors in each processor module), as disclosed by Kimura, as modified by Dupont, Yamagami and Lane, so as said apparatus to be used for an missile tracking with the advantage of providing a secondary track link being

capable of tracking through battlefield conditions and including conventional algorithms to prevent jamming (See Sand, col. 4, lines 41-52).

Referring to claim 9, Sand teaches said sensor interface (i.e., Video Thermal Tracker interface 70 in Fig. 2) comprises

- a processor (i.e., controller 188 of Fig. 2) for processing sensor data (See col. 7, line 66 through col. 8, line 6 and lines 46-57).

Referring to claim 10, Sand teaches said sensor interface (i.e., Video Thermal Tracker interface 70 in Fig. 2) comprises

- a bus interface device (i.e., Sample & Hold 1...N, AGC 1...N, Offset 1...N Correction, and Low Pass Filter 1...N in Fig. 2) for communicating data (i.e., video signal) between said processor (i.e., controller 188 of Fig. 2) and said system bus (i.e., channel 1, 2, ... N in Fig. 2).

Referring to claim 11, Sand teaches said sensor interface (i.e., Video Thermal Tracker interface 70 in Fig. 2) comprises

- a local processor bus (i.e., connecting bus between said controller 188 and Sample & Hold 1...N in Fig. 2) for communicating data (i.e., control signal) between said processor (i.e., controller 188 of Fig. 2) and said bus interface device (i.e., Sample & Hold 1...N, AGC 1...N, Offset 1...N Correction, and Low Pass Filter 1...N in Fig. 2).

5. Claims 19-26 and 28-31 are rejected under 35 U.S.C. 103(a) as being unpatentable over Kimura [JP 409022380 A] in view of Dupont [US 6,842,800 B2], Yamagami [JP 408272756 A]

and Lane [US 5,502,718 A] as applied to claims 1-7 and 14-18 above, and further in view of PCI System Architecture [PCI System Architecture, 3rd Ed., published by Mind Share, Inc. in 1995; hereinafter PCI_System].

Referring to claim 19, Kimura, as modified by Dupont, Yamagami and Lane, discloses all the limitations of the claim 19, except that does not teach requesting a bus grant to transmit data packets to said device; receiving a bus grant signal in response to said step of requesting, indicating that data may be transmitted over a system bus; and transmitting data packets in response to said step of receiving.

PCI_System discloses a PCI Local Bus (See Chapter 3. Introduction to PCI Bus Operation), wherein

- arbitrating steps (i.e., Arbiter for Arbitration Algorithm; See Chapter 6. PCI Bus Arbitration) comprises requesting a bus grant (i.e., asserting REQ#) to transmit data packets to said device (i.e., target device in case of writing operation; See page 85, step 1);
- receiving a bus grant signal (i.e., sampling GNT#) in response to said step of requesting, indicating that data may be transmitted over a system bus (i.e., over PCI bus; See page 86, step 2); and
- transmitting data packets (i.e., beginning a data transaction) in response to said step of receiving (See page 86, step 6).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to have included said arbitrating steps (i.e., PCI arbitration algorithm), as disclosed by PCI_System, in said step of arbitrating, as disclosed by Kimura, as modified by Dupont, Yamagami and Lane, for the advantage of allowing bus arbitration to take place while the bus

owner (i.e., current initiator) is performing a data packets transfer (See PCI_System, page 82, Hidden Bus Arbitration).

Referring to claim 20, PCI_System teaches

- said steps of requesting and receiving are accomplished by a device (i.e., master device) connected to said system bus (i.e., PCI bus; See Fig. 6-1 on page 78 and Fig. 6-3 on page 88).

Referring to claim 21, PCI_System teaches

- said bus grant signal (i.e., GNT#) is issued by a system bus arbitration unit (i.e., PCI Arbiter in Fig. 6-1 on page 78; See page 77, Arbiter).

Referring to claim 22, Kimura, as modified by Dupont, Yamagami and Lane, discloses all the limitations of the claim 22, except that does not teach inquiring if said system bus is in use; verifying that a destination device is not busy once said system bus is not in use; requesting access to said system bus to a system bus arbitration unit; gaining access to said system bus from said system bus arbitration unit; and transmitting data packets to said destination device.

PCI_System discloses a PCI Local Bus (See Chapter 3. Introduction to PCI Bus Operation), wherein

- arbitrating steps (i.e., Arbiter for Arbitration Algorithm; See Chapter 6. PCI Bus Arbitration) comprises inquiring if said system bus (i.e., PCI bus) is in use (i.e., checking if the PCI bus is not in idle state, viz., FRAME# or IRDY# is asserted; See page 86, step 2);

- verifying that a destination device (i.e., target device) is not busy (i.e., TRDY# is asserted) once said system bus is not in use (i.e., once the PCI bus is in idle state, viz., both of FRAME# and IRDY# are deasserted; See page 86, steps 2-6);
- requesting access (i.e., asserting REQ#) to said system bus to a system bus arbitration unit (i.e., PCI Arbiter in Fig. 6-1 on page 78; See page 77, Arbiter);
- gaining access to said system bus from said system bus arbitration unit (See page 86, step 5, and Fig. 6-3 on page 88); and
- transmitting data packets (i.e., beginning a data transaction) to said destination device (i.e., target device; See page 86, step 6).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to have included said arbitrating steps (i.e., PCI arbitration algorithm), as disclosed by PCI_System, in said step of arbitrating, as disclosed by Kimura, as modified by Dupont, Yamagami and Lane, for the advantage of allowing bus arbitration to take place while the bus owner (i.e., current initiator) is performing a data packets transfer (See PCI_System, page 82, Hidden Bus Arbitration).

Referring to claim 23, PCI_System teaches

- said system bus arbitration unit (i.e., PCI Arbiter in Fig. 6-1 on page 78) allows continual access to said system bus (i.e., performing additional transactions upon completion of the current transaction) if said destination device does not become busy, if said bus does not become busy, and if no other device requests access to said system bus (See page 85, lines 21-28).

Referring to claim 24, PCI_System teaches

- said system bus arbitration unit (i.e., PCI Arbiter in Fig. 6-1 on page 78) grants access to a second device (i.e., a bus request device to perform a next transaction) upon request during a transmission of a data packet (i.e., bus arbitration being taken place while the current transaction is performed by an initiator) by another device (i.e. the initiator of the current transaction) on said system bus (See Hidden Bus Arbitration, on page 82).

Referring to claim 25, PCI_System teaches

- access to said system bus (i.e., accessing to PCI bus) is granted to a second device (i.e., a PCI master device performing the next transaction) by said system bus arbitration unit (i.e., PCI Arbiter in Fig. 6-1 on page 78), which executes the steps of: discontinuing bus grant access to any device currently transmitting data (i.e., deasserting GNT# from the bus master currently transmitting data; See page 86, step 4);
- verifying that said system bus is not busy (i.e., checking if the PCI bus is in idle state, viz., both of FRAME# and IRDY# are deasserted);
- verifying that a destination device (i.e., target device) is not busy (i.e., TRDY# is asserted; See page 86, steps 2-6);
- granting access to said system bus for said second device requesting access (See page 86, step 5);
- delaying any further transmission by said device whose access to said system bus was discontinued by said step of discontinuing until after at least one data packet has been transmitted by said second device (See page 86, step 7-16 and page 82, Hidden Bus Arbitration).

Referring to claim 26, PCI_System teaches

- access to said system bus between multiple devices connected to said system bus is granted according to priority (See page 79, lines 5-7).

Referring to claim 28, Kimura teaches

- devices (i.e., processor modules 1 in Fig. 1) connected to said system bus (i.e., system bus 2 of Fig. 1) contain local and module busses (i.e., a bus between processor 3 and proper cache 4, and an intramodule bus 5 in Fig. 1) connected to said system bus by way of a gateway device (i.e., connected to said system bus 2 via system interface controller 6 in Fig. 1), which arbitrates access to nodes (i.e., a plural pairs of respective processor 3 and proper cache 4 in Fig. 1) connected to said module bus (See Solution on the Brief Summary).

Referring to claim 29, Kimura, as modified by Dupont, Yamagami and Lane, discloses all the limitations of the claim 29, including said gateway device (i.e., system interface controller 6 of Fig. 1; Kimura) arbitrates access to said local and module busses (See Kimura, Solution on the Brief Summary), except that does not teach said arbitrating is performed according to priority.

PCI_System discloses a PCI Local Bus (See Chapter 3. Introduction to PCI Bus Operation), wherein

- arbitrating steps (i.e., Arbiter for Arbitration Algorithm; See Chapter 6. PCI Bus Arbitration) is performed according to priority (See page 79, lines 5-7).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to have included said arbitrating steps (i.e., PCI arbitration algorithm), as disclosed by PCI_System, in said step of arbitrating, as disclosed by Kimura, as modified by Dupont,

Yamagami and Lane, for the advantage of providing a fairness algorithm to avoid deadlocks (See PCI_System, page 79, lines 11-20).

Referring to claim 30, Kimura, as modified by Dupont, Yamagami, Lane and PCI_System, teaches

- said gateway device (i.e., system interface controller 6 of Fig. 1; Kimura) arbitrates access to said local and module busses (i.e., a bus between processor 3 and proper cache 4, and an intramodule bus 5 in Fig. 1; Kimura) in a rotating fashion (See PCI_System, page 80, lines 20-22).

Referring to claim 31, Kimura, as modified by Dupont, Yamagami and Lane, discloses all the limitations of the claim 31, except that does not teach inquiring if said module bus is in use; verifying that a destination processor is not busy once said module bus is not in use; requesting access to said module bus to a bus gateway device; gaining access to the module bus from said bus gateway device; and transmitting data packets to said destination processor.

PCI_System discloses a PCI Local Bus (See Chapter 3. Introduction to PCI Bus Operation), wherein arbitrating steps (i.e., Arbiter for Arbitration Algorithm; See Chapter 6. PCI Bus Arbitration) comprises

- inquiring if said module bus (i.e., PCI bus) is in use (i.e., checking if the PCI bus is not in idle state, viz., FRAME# or IRDY# is asserted; See page 86, step 2);
- verifying that a destination processor (i.e., target device) is not busy (i.e., TRDY# is asserted) once said module bus is not in use (i.e., once the PCI bus is in idle state, viz., both of FRAME# and IRDY# are deasserted; See page 86, steps 2-6);

- requesting access to said module bus to a bus gateway device (i.e., PCI Arbiter in Fig. 6-1 on page 78; See page 77, Arbiter);
- gaining access to the module bus from said bus gateway device (See page 86, step 5, and Fig. 6-3 on page 88); and
- transmitting data packets (i.e., beginning a data transaction) to said destination processor (i.e., target device; See page 86, step 6).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to have included said arbitrating steps (i.e., PCI arbitration algorithm), as disclosed by PCI_System, in said method for managing a flow of information, as disclosed by Kimura, as modified by Dupont, Yamagami and Lane, for the advantage of allowing bus arbitration to take place while the bus owner (i.e., current initiator) is performing a data packets transfer (See PCI_System, page 82, Hidden Bus Arbitration).

6. Claim 27 is rejected under 35 U.S.C. 103(a) as being unpatentable over Kimura [JP 409022380 A] in view of Dupont [US 6,842,800 B2], Yamagami [JP 408272756 A], Lane [US 5,502,718 A] and PCI_System as applied to claims 19-26 and 28-31 above, and further in view of McDonald et al. [US 6,138,176 A; hereinafter McDonald].

Referring to claim 27, Kimura, as modified by Dupont, Yamagami, Lane and PCI_System, discloses all the limitations of the claim 27 including access to said system bus between multiple devices connected to said system bus is granted in a rotating fashion based on said priority (See PCI_System, page 80, lines 20-22), except that does not teach said rotating fashion for a maximum of a time required to transfer one data packet.

McDonald discloses a high-performance RAID system (See Abstract), wherein

- access to a system bus (i.e., packet-switched bus) between multiple devices (i.e., automated controllers AC_{1-8} 84 in Fig. 6) connected to said system bus is granted (i.e., granting time slots on said packet-switched bus) in a rotating fashion (i.e., round robin protocol) for a maximum of a time required to transfer one data packet (See col. 18, lines 23-27).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to have included said method step of accessing system bus, as disclosed by McDonald, in said step of arbitrating, as disclosed by Kimura, as modified by Dupont, Yamagami, Lane and PCI_System, for the advantage of obviating a requirement of suspending said destination device read or write operation (i.e., disk read or disk write operation) as the result insufficient bandwidth on said system bus (i.e., packet-switched bus; See McDonald, col. 18, lines 35-38).

Response to Arguments

7. Applicant's arguments filed on 15th of March 2006 with respect to claims 1 and 17 have been considered but are moot in view of the new ground(s) of rejection.

In fact, the arguments with the new issue are drawn to the newly added limitations which have not been considered. Thus, the Applicant's argument on this point is moot in view of further consideration requirement.

However, the newly added limitations "a plurality of processors, each processor being in communication with a respective local processor bus" and "control information packets being separately buffered from other packets in a buffer separately connected between the respective local processor bus and the system bus" in the exemplary claim 1 are still suggested by Kimura and Dupont, respectively. See paragraph 3 of the instant Office Action, Claims 1-7 and 14-18

rejection under 35 U.S.C. 103(a) as being unpatentable over Kimura in view of Dupont, Yamagami and Lane.

Conclusion

8. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Christopher E. Lee whose telephone number is 571-272-3637. The examiner can normally be reached on 9:30am - 5:30pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Rehana Perveen can be reached on 571-272-3676. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Christopher E. Lee
Patent Examiner
Art Unit 2112

CEL/

Christopher E. Lee